第 1 章要点——初见网络爬虫

1．安装 python3.5：将目录选在 C:\Python35

2．HTML 标签（以 http://www.pythonscraping.com/pages/page1.html 为例）

- 大多数 HTML 元素被定义为块级元素（block level element）或内联元素（inline element）
- 块级元素在浏览器显示时，通常会以新行来开始（和结束）。如：<h1>, <p>, <ul>, <table>
- 内联元素在显示时通常不会以新行开始。如：<b>, <td>, <a>, <img>

3．urllib 库包括 urllib.request、urllib.parse、urllib.error 三个子模块

```
from urllib.request import urlopen
html=urlopen("http://www.pythonscraping.com/pages/page1.html")
print(html.read())
```

4．BeautifulSoup 库

- 通过定位 HTML 标签来格式化和组织复杂的网络信息，它不是标准库
- 安装 BeautifulSoup：pip install beautifulsoup4

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
def get_title(url):
    html=urlopen(url)
    bsObj=BeautifulSoup(html.read(),"html.parser")
    return bsObj.html.body.h1
title=get_title("http://www.pythonscraping.com/pages/page1.html")
print(title)
```

5．网页打不开的情形

```
from urllib.error import URLError
from urllib.request import urlopen
from bs4 import BeautifulSoup
def get_title(url):
    try:
        html=urlopen(url)
        bsObj=BeautifulSoup(html.read(),"html.parser")
        return bsObj.html.body.h1
    except URLError as e:
        print(e)
title=get_title("http://www.pythonscraping.com/pages/page1.html")
print(title)
```

6．标签找不到的情形

```
from urllib.error import URLError
from urllib.request import urlopen
from bs4 import BeautifulSoup
def get_title(url):
    try:
        html=urlopen(url)
        bsObj=BeautifulSoup(html.read(),"html.parser")
        return bsObj.html.body.h1
    except URLError as e:
        print(e)
    except AttributeError as e:
        print(e)
title=get_title("http://www.pythonscraping.com/pages/page1.html")
if title==None:
    print("Title could not be found")
else:
    print(title)
```